Number of instructions in 8085 Microprocessor

INTEL 8085 has a very enriched Instruction Set. Varieties of instructions it can execute. Instructions will have different Byte counts, ranging from 1-Byte to 3-Bytes. Opcode always occupies 1-Byte in the memory. As we know that, with 8 bits for the opcode, 2^8 = 256 distinct opcodes are possible. In hexadecimal notation, the opcodes can range from 00H to FFH. Each opcode will correspond to an instruction. Thus from the calculation, it is possible to have 256 instructions in the instruction set of 8085. However, only 246 opcodes are implemented in 8085. They can be classified under 66 types, which are broadly classified into the six groups as depicted below –

Description	**No. of opcodes**	**No. of instruction types**	**Example**
Data transfer Instructions	83	13	MOV, MVI etc.
Arithmetic Instructions	62	14	ADD, SUB etc.
Logical Instructions	43	15	AND, XOR etc.
Stack Instructions	15	9	PUSH, POP etc.
Branch Instructions	36	8	JMP, JZ etc.
I/O Instructions	2	2	IN, OUT etc.
Interrupt Instructions	5	5	RST 0, RST 1 etc.
Total	246	66	

Out of these 246 opcodes, we have –

Classifications	**Examples**
202 Opcodes which are 1-Byte long	MOV A, B ANA B
18 Opcodes which are 2-Bytes long	MVI A, d8 ANI d8
26 Opcodes which are 3-Bytes long	JMP d16 LXI B, d16

ROTATE Instructions in 8085

ROTATE is a logical operation of the 8085 microprocessor. It is a 1-byte instruction. This instruction does not require any operand after the opcode. It operates the content of the accumulator and the result is also stored in the accumulator. The Rotate instruction is used to rotate the bits of accumulator. **Types of ROTATE Instruction:** There are 4 categories of the ROTATE instruction: Rotate accumulator left (RLC), Rotate accumulator left through carrying (RAL), Rotate accumulator right (RRC), Rotate accumulator right through carry (RAR). Among these four instructions; two are for rotating left and two are for rotating right. All of them are explained briefly in the following sections:

1. **Rotate accumulator left (RLC) –** In this instruction, each bit is shifted to the adjacent left position. Bit D7 becomes D0. Carry flag CY is modified according to the bit D7. For example:-

A = D7 D6 D5 D4 D3 D2 D1 D0
//before the instruction
A = 10101010; CY=0
//after 1st RLC
A = 01010101; CY=1
//after 2nd RLC
A = 10101010; CY=0

1. **Rotate accumulator left through carry (RAL) –** In this instruction, each bit is shifted to the adjacent left position. Bit D7 becomes the carry bit and the carry bit is shifted into D0. Carry flag CY is modified according to the bit D7. For example:

A = D7 D6 D5 D4 D3 D2 D1 D0
//before the instruction
A = 10101010; CY=0
//after 1st RAL
A = 01010100; CY=1
//after 2nd RAL
A = 10101001; CY=0

1. **Rotate accumulator right (RRC) –** In this instruction, each bit is shifted to the adjacent right position. Bit D7 becomes D0. Carry flag CY is modified according to the bit D0. For example:

A = D7 D6 D5 D4 D3 D2 D1 D0
//before the instruction
A = 10000001; CY=0
//after 1st RRC
A = 11000000; CY=1
//after 2nd RRC
A = 01100000; CY=0

1. **Rotate accumulator right through carry (RAR) –** In this instruction, each bit is shifted to the adjacent right position. Bit D0 becomes the carry bit and the carry bit is shifted into D7. Carry flag CY is modified according to the bit D0. For example:

A = D7 D6 D5 D4 D3 D2 D1 D0
//before the instruction
A = 10000001; CY=0
//after 1st RAR
A = 01000000; CY=1
//after 2nd RAR
A = 10100000; CY=0

Applications of ROTATE Instructions: The ROTATE instructions are primarily used in arithmetic multiply and divide operations and for serial data transfer. For example:

If A is 0000 1000 = 08H
1. By rotating 08H right : A = 0000 0100 = 04H
This is equivalent to *dividing by 2*.
2. By rotating 08H left : A = 0001 0000 = 10H
This is equivalent to *multiplying by 2*.

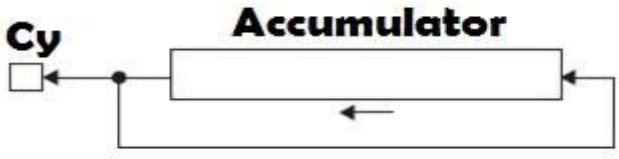
However, these procedures are invalid when logic 1 is rotated left from D7 to D0 or vice versa. For example, if 80H is rotated left it becomes 01H.

Instructions to rotate Accumulator in 8085 Microprocessor

In 8085 Instruction set, there a set of instructions which can Accumulator contents left or right. It is to be noted here that rotate operation can be performed only on Accumulator contents. These instructions set is listed below–

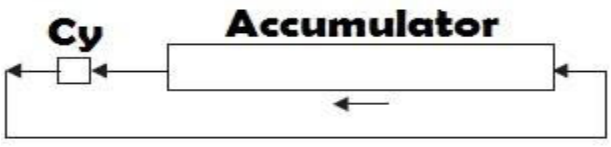
Mnemonics, Operand	Opcode(in HEX)	Bytes
RAL	17	1
RAR	1F	1
RLC	07	1
RRC	0F	1

Mnemonic **RLC** stands for "Rotate Left Accumulator". It rotates the Accumulator contents to the left by 1-bit position. The following Fig. shows the operation explicitly –



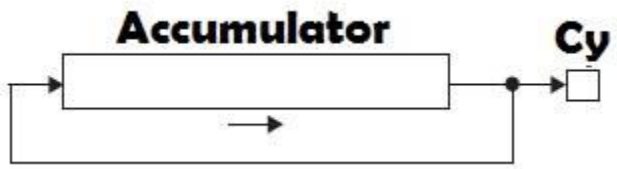
In this fig. it has been depicted that the most significant bit of the Accumulator will come out and left rotate will create an empty space at the least significant bit place and this come out bit will be copied at the empty bit place and also on the Cy bit in the flag register. Thus, Cy flag gets a copy of the bit moved out from the MS bit position. Notice that Cy flag is not involved in the rotation, and it is only 8-bit rotation of Accumulator contents. Only Cy flag is affected by this instruction execution.

Mnemonic **RAL**, which stands or Rotate Accumulator Left and also involving Cy flag in rotation. It rotates the Accumulator contents to the left by 1-bit position. The following Fig. is depicting the execution logic of the instruction.



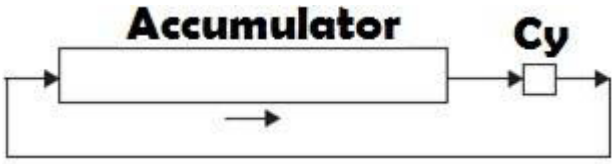
From the Fig. we can see that the due to left rotation the bit which is coming out from the most significant place will be copied to the Cy flag bit. And the previous Cy bit will be moved to least significant bit place of the Accumulator. Thus it is a 9-bitrotation of Accumulator and Cy contents. Only Cy flag is affected by this instruction execution.

Mnemonic **RRC** stands for "Rotate Right Accumulator". With the help of this instruction, we can rotate the Accumulator current content to the right by 1-bit position. The following Fig. will depict the rotation operation –



In this right rotation, the least significant bit will come out from the Accumulator and will be copied to Cy bit in the flag register and also will be copied to the most significant bit position of the Accumulator. Notice that the Cy flag is not involved in the rotation, and it is only 8-bit rotation of Accumulator contents. Only Cy flag is affected by this instruction execution.

Mnemonic **RAR** stands for "Rotate Accumulator Right involving Cy flag in rotation". It rotates the Accumulator contents to the right by1-bit position. From the following Fig. we are getting the operation details –



From the Fig. we can see that, during right rotate the least significant bit is coming out and will be copied on the Cy flag bit and the previous Cy flag bit will be moved to the most significant bit position of the Accumulator. It is 1-Byteinstruction. And it is 9-bit rotation of Accumulator and Cy contents. Only Cy flag is affected by this instruction execution.